Digital Image Processing Course Report

Improve visual realism using digital image processing techniques for landing pad

---

Course Information

- Course Name: Digital Image Processing
- Academic Year: 2024 Fall Semester

Student Information

- Name: 段瑞豪
- Student ID: 2022180163
- Major: Computer Science and Technology
- Class: [Your Class]

Instructor

Dr. Qingfeng Zhang （张庆丰）

Submission Information

- Submission Date: 2024/12/25
- Report Type: Project

International School, Jinan University

Contents

1. Abstract

Visual realism is a critical component in simulation environments, particularly in flight simulation systems. Enhancing the textures of key elements, such as landing pads, plays a vital role in achieving realism and improving user immersion. This report explores the application of digital image processing techniques to enhance the visual realism of landing pad textures. The approach employs grayscale conversion, blurring, and edge detection to emphasize structural details and improve visual clarity. The results demonstrate the potential of these techniques for simulation systems.

2. Introduction

In modern flight simulation systems, visual fidelity is a crucial factor that impacts user experience and situational awareness. For elements like landing pads, which serve as visual references for pilots during landing operations, enhancing texture details can significantly improve the authenticity of the simulation. This report presents a systematic approach to improving the visual realism of a landing pad texture using digital image processing techniques, with a specific focus on enhancing edge details and overlaying them onto the original image for a sharper and more realistic representation.

3. Methodology

The methodology involves several sequential image processing steps applied to the landing pad texture. These steps include:

Step 1: Grayscale Conversion

Converting the image to grayscale reduces computational complexity by eliminating color information while retaining structural details. Grayscale images are often used as the base for further image processing techniques such as edge detection.

Step 2: Gaussian Blurring

Gaussian blur is applied to soften the image and reduce noise. This step ensures that small, irrelevant details are smoothed out, making subsequent edge detection more focused on significant features.

Step 3: Edge Detection

Edge detection highlights the structural features of the landing pad, such as boundaries, markings, and other distinct patterns. The Canny edge detection algorithm is used, as it is robust and provides fine control over the detection thresholds.

Step 4: Overlaying Edges

The extracted edges are overlaid onto the original image to emphasize critical details. By blending the edge map with the original texture, we create an enhanced texture that retains the original image's color information while amplifying structural features.

Implementation

The implementation uses Python and the OpenCV library for image processing. Below is the code snippet illustrating the process:

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Load the landing pad texture
image =
cv2.imread("C:\\Users\\Duanm\\FlightSimulationCore\\ML_Predictions\\Helicopter
Landing Pad Stock Photos and Images - 123RF.jpeg")

# Check if the image loaded correctly
if image is None:
    raise ValueError("Image not found or could not be loaded.")

# Step 1: Convert to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

# Step 2: Apply GaussianBlur to soften
blurred = cv2.GaussianBlur(gray, (5, 5), 0)
```

```
# Step 3: Use edge detection to highlight markings
edges = cv2.Canny(blurred, 50, 150)

# Step 4: Overlay edges on the original texture
edges_colored = cv2.cvtColor(edges, cv2.COLOR_GRAY2BGR)  # Convert edges to BGR
enhanced_texture = cv2.addWeighted(image, 0.8, edges_colored, 0.2, 0)

# Save the enhanced texture
cv2.imwrite("enhanced_landing_pad.png", enhanced_texture)

# Display the result
plt.imshow(cv2.cvtColor(enhanced_texture, cv2.COLOR_BGR2RGB))
plt.title("Enhanced Landing Pad Texture")
plt.axis("off")
plt.show()
```

4. Results

The application of the outlined digital image processing techniques yields an enhanced landing pad texture. The key results are:

1. **Improved Edge Visibility:** The Canny edge detection algorithm successfully highlights critical features such as landing pad markings and boundaries.
2. **Blended Enhancement:** By overlaying the edges onto the original image with weighted blending, the final texture retains its natural appearance while emphasizing structural details.
3. **Visual Realism:** The enhanced texture provides improved clarity, making it more visually engaging and realistic for simulation environments.

The transition from the original image of the landing pad, as captured in a real-world environment, to the enhanced 3D representation involves multiple stages of digital image processing and simulation techniques. Initially, the original landing pad image undergoes preprocessing using computer vision techniques, such as grayscale conversion, Gaussian blurring for texture refinement, and edge detection to emphasize critical markings like the "H" symbol and boundary lines. These processed elements are combined to create an enhanced texture that maintains clarity and highlights essential features of the landing pad.

The refined texture is then imported into Unity, where it is mapped onto a 3D terrain generated using Perlin noise to simulate the surrounding environment. The integration process involves assigning the enhanced texture to a material in Unity and applying it to the landing pad model within the simulation scene. This step bridges the gap between real-world imagery and virtual representation. The final result is a realistic and immersive depiction of the landing pad, placed within a procedurally generated terrain that mimics a real-world setting. The comparison between the original and the Unity visualization underscores the effectiveness of combining image processing and rendering techniques to achieve high levels of realism.

The enhanced landing pad texture, *enhanced_landing_pad.png*, was integrated into the simulation environment using Unity to improve the visual realism of the landing pad. The texture was first imported into Unity's Assets folder and utilized to create a new material, designated as *LandingPadMaterial*. This material was configured by assigning the enhanced texture to the *Albedo* property, ensuring accurate representation of the processed design. The material was then applied to the 3D landing pad model within the scene, enabling the detailed and visually enhanced texture to be rendered effectively. This process allowed the simulation to achieve a higher level of detail and realism for the landing pad, improving both its aesthetic and functional representation.

The outputs of the process are shown as follows:

Figure: Original view of the landing pad on a heliport

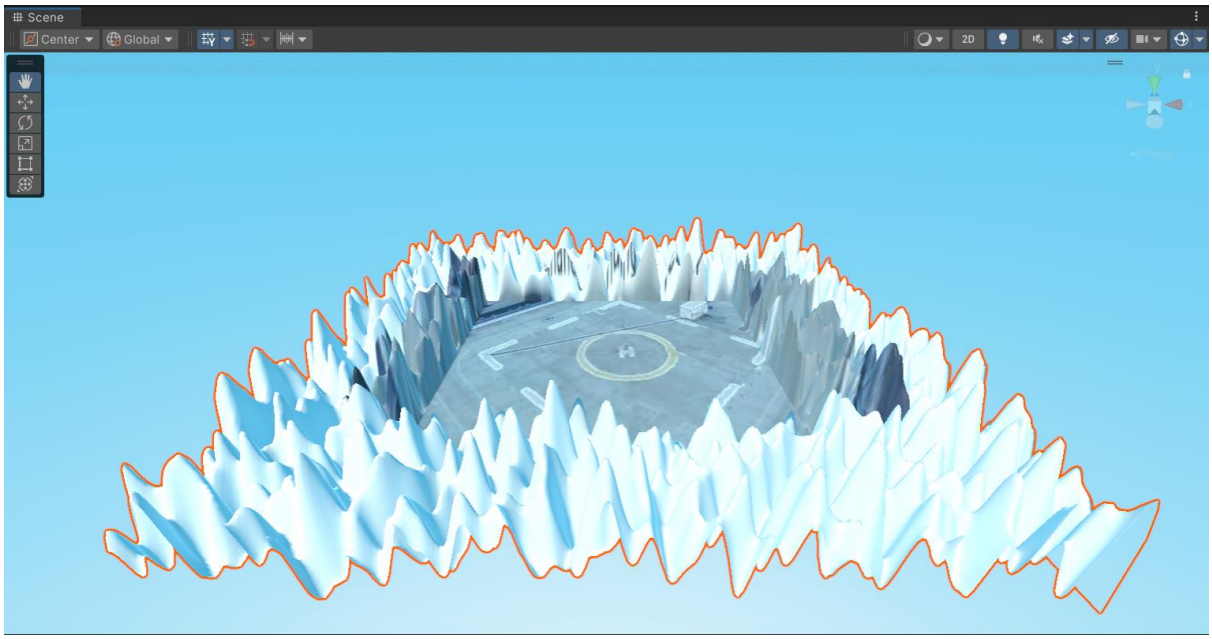

Figure: Realistic view of the “Harbin’s” landing pad visualized in Unity

Figure: Original view of the landing pad on a skyscraper

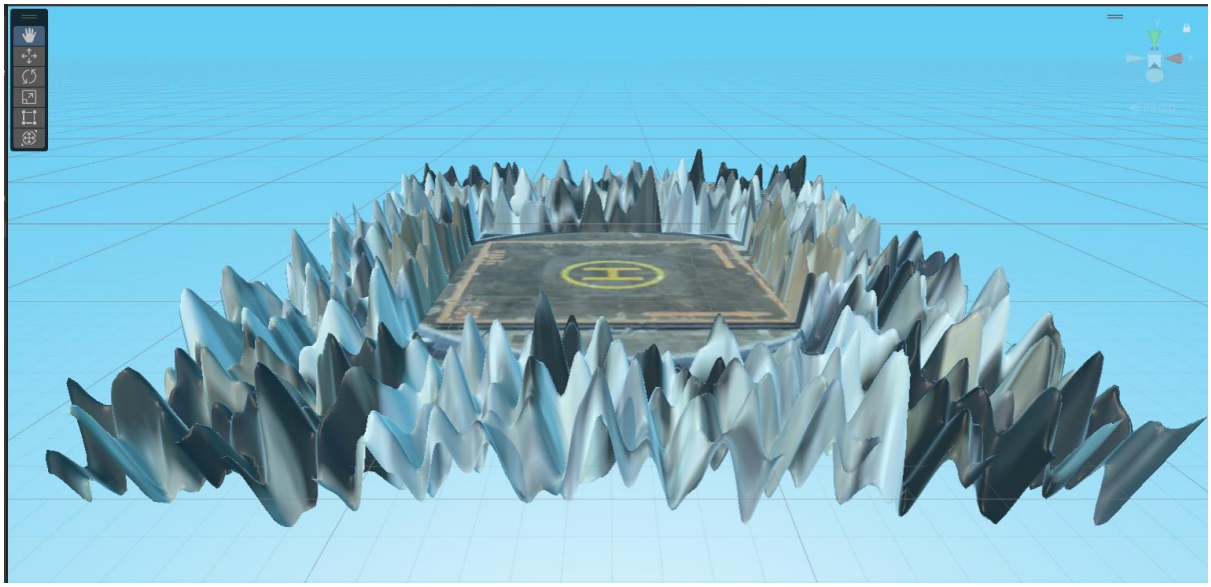

Figure: Realistic view of the an abandoned landing pad visualized inside the forest in Unity

Figure: Original view of the landing pad on shore

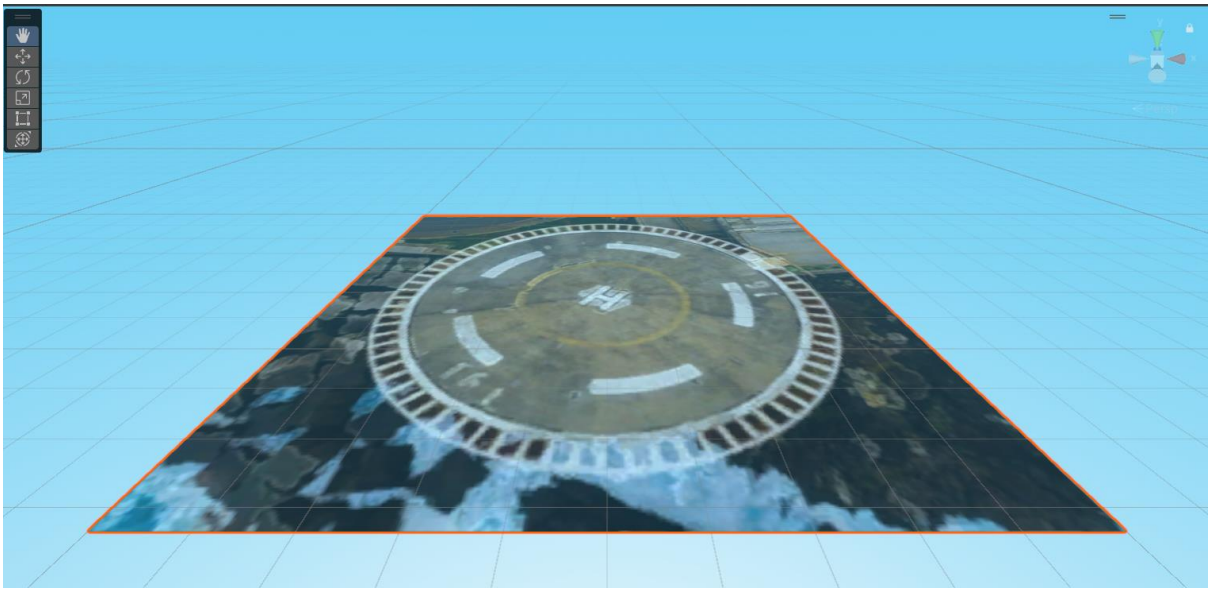

Figure: Realistic view of a pilot approaching a landing pad visualized in Unity

Figure: Original view of a helicopter landing on a ship

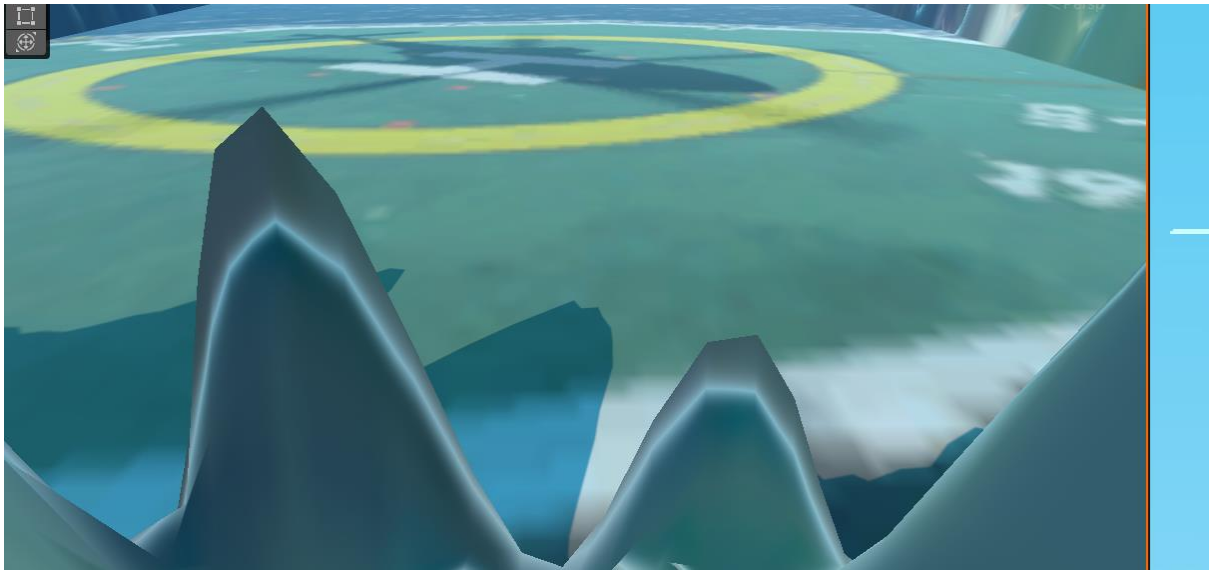

Figure: Close Up view of the landing pad visualized in Unity

4.2 Discussion

The use of digital image processing techniques allows for a systematic enhancement of textures without the need for manual editing. This approach is highly scalable and can be applied to other simulation elements. However, the effectiveness of the enhancement depends on the quality of the input texture and the tuning of parameters such as the Gaussian blur kernel size and Canny thresholds.

4.3 Limitations

1. **Input Texture Quality:** Low-resolution textures may not benefit significantly from edge enhancement.
2. **Parameter Sensitivity:** The choice of parameters (e.g., blur kernel size, edge detection thresholds) requires careful tuning to avoid over-enhancement or loss of detail.
3. **Real-Time Integration:** While effective for pre-processing, the computational cost of these techniques may limit their use in real-time applications.

5. Conclusion

This report demonstrates the application of digital image processing techniques to improve the visual realism of landing pad textures for flight simulation systems. By leveraging grayscale conversion, Gaussian blurring, edge detection, and blending, significant improvements in texture clarity and detail are achieved. These techniques provide a robust framework for enhancing visual realism in simulation environments, contributing to a more immersive and authentic user experience.

5.1 Future Work

1. **Real-Time Processing:** Investigate methods to optimize these techniques for real-time application in simulation engines.
2. **Advanced Enhancements:** Explore the use of machine learning models for texture enhancement and feature recognition.
3. **Integration with 3D Models:** Extend the methodology to enhance 3D terrain models and other simulation assets.

6. References

1. OpenCV Documentation: https://docs.opencv.org
2. Canny, J. (1986). "A Computational Approach to Edge Detection." IEEE Transactions on Pattern Analysis and Machine Intelligence.
3. Szeliski, R. (2022). *Computer Vision: Algorithms and Applications* (2nd ed.). Springer.
4. Unity Technologies. (2024). *Unity Manual: Materials, Shaders, and Textures*.
5. Tavinor, G. (2009). *The Art of Videogames*. Wiley-Blackwell.
6. Ebert, D. S., Musgrave, F. K., Peachey, D., Perlin, K., & Worley, S. (2003). *Texturing & Modeling: A Procedural Approach* (3rd ed.). Morgan Kaufmann.
7. Luo, R. C., & Kay, M. G. (1989). Multisensor integration and fusion for intelligent systems. *IEEE Transactions on Systems, Man, and Cybernetics, 19*(5), 901–931.